

245990US

TITLE OF THE INVENTION

METHOD AND APPARATUS FOR COMPENSATING FOR MOTION PREDICTION

5

BACKGROUND OF THE INVENTION

Technical Field of the Invention

This invention relates to a method and apparatus for compensating for motion prediction, which is suitably applicable to an image coding apparatus for transmitting, for example, motion picture signals over networks, such as the Internet.

Description of the Related Art

An image coding apparatus, conventionally, is configured to digitize a motion image signal supplied from an external source and then carries out code processing on it conforming to a predetermined image-coding scheme, thereby generating image-compressed information.

The foregoing image-coding scheme is a known image coding scheme, called MPEG, standardized for the purpose of coding a general-use image by the Moving Picture Experts Group (MPEG) of ISO/IEC, and the image-coding scheme, called H.26* (i.e., H.261, H.262, ...), standardized aiming at coding a television-conference image.

Meanwhile, the recent widespread use of personal digital assistants (PDA's) including cellular phones gives rise to the need for an image-coding scheme to realize still higher coding

efficiency. In order to cope with this, the standardization of another image-coding scheme, called JVT (Joint Model of Enhanced-Compression Video Coding) (hereinafter, this is called a JVT coding scheme), is currently being pushed forward by the
5 MPEG and ITU entities.

Under the JVT coding scheme, when carrying out motion-prediction compensation, motion vectors can be searched over the macro-block having vertically horizontally 16×16 pixels with the use of four kinds of size patterns (hereinafter, this
10 is referred to as a macro-block mode) TP1 - TP4, e.g. four sets of motion compensating blocks comprising one set of pixel blocks having a size of vertical/horizontal 16×16 pixels (hereinafter, this is referred to as motion compensating blocks), two sets of motion compensating blocks each having a size of
15 vertical/horizontal 8×16 pixels, two sets of motion compensating blocks each having a size of vertical/horizontal 16×8 pixels, or four sets of motion compensating blocks each having a size of vertical/horizontal 8×8 pixels, as shown in Fig. 16. Each of the motion compensating blocks related to these macro-block
20 modes TP1 - TP4 can independently have the motion vectors.

Furthermore, concerning the four sets of motion compensating blocks related to the macro-block mode TP4, motion vectors can be searched on each of the four sets of the compensating blocks by use of four kinds of sub-size patterns (hereinafter,
25 this is referred to as a sub-macro block mode) TP5 - TP8, e.g.

one set of motion compensating blocks having a size of vertical/horizontal 8x8 pixels, two sets of motion compensating blocks each having a size of vertical/horizontal 4x8 pixels, two sets of motion compensating blocks each having a size of vertical/horizontal 8x4 pixels, or four sets of motion compensating blocks each having a size of vertical/horizontal 4x4 pixels. Each of the motion compensating blocks related to these sub-macro block modes can independently have the motion vectors.

10 Accordingly, under the JVT coding scheme, one macro-block can have 16 motion vectors in maximum (see, for example, Draft ISO/IEC 14496 - 10:2002(e)), the entire contents of which being incorporated herein by reference.

 Meanwhile, under the JVT coding scheme, when compensating
15 for motion prediction, it is possible to hold a group SF of a plurality of reference frame images. It is possible to carry out a block matching by using two or more reference frame images SF1, SF2 and SFn-1 existing prior (past) to an objective frame image OF to be code-processed, a block matching by using two
20 or more reference frame images (not shown) existing subsequent (future) to an objective frame image OF, or a block matching by using two or more reference frame images (not shown) existing at around an objective frame image OF. The group SF of reference frame images held as the referential candidates is referred to
25 as a Multiple Reference Frame (see, for example, Draft ISO/IEC

14496 - 10:2002(e)). Furthermore, the JVT coding scheme adopts optimized motion-prediction compensation called Rate-Distortion Optimization (hereinafter, this is referred to as RD optimization). By RD optimization, all the macro-blocks of an objective frame image OF can be block-matched with the group SF of a plurality of reference frame images (Fig. 16), thereby determining a difference value (distortion) in a difference absolute-value sum or the like. Taking into account a quantizing function (Lagrange multiplier for motion vectors) and generation code amount of motion vector difference, a motion vector is searched for which can generate a minimum code amount, thereby enhancing the coding efficiency for motion vectors See DRAFT ISO/IEC 14496-10: 2002(E), and

Rate-Distortion Optimization for Video Compression (G. Sullivan and T. Wiegand, IEEE Signal Processing Magazine, 1998-11), the entire contents of which being incorporated herein by reference.

Meanwhile, in the coding apparatus employing the foregoing JVT coding scheme, while all the macro-blocks are changed to motion compensating blocks in various pixel-based sizes (Fig. 16) between the macro-block entirety and a group SF of a plurality of reference frame images, reference is made to a generation code amount table representative of a generated code amount of motion vector difference and further a quantizing function (Lagrange multiplier for motion vectors) is determined

besides a difference value (distortion), thereby carrying out a motion-prediction compensation. Thus, the present inventors have recognized that there is a drawback that process burden is increased due to code processing.

5

SUMMARY OF THE INVENTION

It is an object of the present invention to provide a method and apparatus for compensating for motion prediction which can reduce the process burden due to code processing while possibly
10 avoiding the lowering image quality.

In order to solve the foregoing and other problems, in the present invention, when compensating for motion prediction on each motion compensating blocks formed by dividing an objective frame image
15 of successive frame images by using a plurality of reference frame images while sequentially changing pixel-based sizes of the motion compensating blocks. Pixels are thinned out of the motion compensating block having a greatest one of the pixel-based size to be taken as an uppermost layer of among the
20 pixel-based sizes, to generate a size-reduced block in a lower layer having a predetermined size-reduction ratio. Thereafter, motion vector search ranges are determined respectively within the reference frame images, on the basis of the size-reduced blocks and a plurality of size-reduced reference images reduced
25 in size correspondingly respectively to the size-reduction

ratios of the size-reduced blocks. An optimal one of the motion vectors is detected while sequentially changing the pixel-based sizes of the motion compensating blocks by using the search range determined.

5 In this case, motion vectors can be detected only in a limited search range of each reference frame image without using the entire of the reference frame images.

BRIEF DESCRIPTION OF THE DRAWINGS

10 Fig. 1 is a schematic diagram for explaining the motion compensation at 1/4-pixel accuracy under the JVT coding scheme;

 Fig. 2 is a schematic diagram for explaining the motion compensation at 1/8-pixel accuracy under the JVT coding scheme;

 Fig. 3 is a schematic diagram showing the motion
15 compensation at 1/8-pixel accuracy using pixel-based values on a diagonal line;

 Fig. 4 is a schematic diagram for explaining the motion compensation in chrominance signals;

 Fig. 5 is a block diagram showing a configuration of an
20 image coding apparatus according to the present invention;

 Fig. 6 is a schematic diagram for explaining a thinning-out process on a field;

 Fig. 7 is a block diagram showing a configuration of a motion-prediction compensating section;

25 Fig. 8 is a schematic diagram showing a manner of

hierarchization;

Fig. 9 is a flowchart showing a motion-prediction compensating process procedure;

Fig. 10 is a schematic diagram for explaining an example
5 to search for a motion vector;

Fig. 11 is a schematic diagram for explaining a first determining example on a motion vector search range;

Fig. 12 is a schematic diagram for explaining a detecting sequence of motion vectors;

10 Fig. 13 is a schematic diagram for explaining a second determining example on a motion vector search range;

Fig. 14 is a schematic diagram for explaining a motion vector detecting sequence in another embodiment;

Fig. 15 is a schematic diagram for explaining a block
15 matching according to another embodiment of the present invention;

Fig. 16 is a schematic diagram showing a motion compensating block as defined under the JVT coding scheme; and

Fig. 17 is a schematic diagram for explaining a Multiple
20 Reference Frame as defined under the JVT coding scheme.

DETAILED DESCRIPTION OF THE INVENTION

JVT Coding Scheme Overview

At the outset, explanation is made on the motion
25 compensation as defined under the JVT coding scheme. Under the

JVT coding scheme, motion compensation is available at 1/4-pixel accuracy and at 1/8-pixel accuracy. The motion compensation process at 1/4-pixel accuracy (interpolation process) is explained with reference to Fig. 1.

5 The JVT coding scheme uses a FIR (Finite Impulse Response) filter as a band limiting filter, to compensate for motion. As a filter coefficient for generating a pixel-based value (luminance signal) at 1/2-pixel accuracy, there is defined a 6-tap filter coefficients as follows.

$$\{ 1, -5, 20, 20, -5, 1 \} \quad \dots\dots (1)$$

10

Herein, as for the motion compensation (interpolation) for the pixel-based value b, h shown in Fig. 1, sum-of-products operation is made by the use of the filter coefficients of vector(1), according to the following equation.

$$b = (E - 5 F + 20 G + 20 H - 5 I + J)$$

$$h = (A - 5 C + 20 G + 20 M - 5 R + T) \quad \dots\dots (2)$$

15

Then, provided that the 8 bits of from "0" to "255" are considered as Clip 1, computation is to be made by a clip process to remove the other part than Clip 1 as given in the following equation.

$$b = \text{Clip1}((b + 16) >> 5)$$

$$h = \text{Clip1}((h + 16) >> 5) \quad \dots\dots (3)$$

20

In this connection, ">>5" in Equation (3) means to shift

by 5 bits, wherein actually division is in an amount of 2^5 .
 Meanwhile, "+16" in Equation (3) is in order not to remove
 fractions (not to be rounded) when division is made in an amount
 of 2^5 .

5 Meanwhile, concerning the pixel-based value j , a
 pixel-based values $aa, bb, cc, dd, ee, ff, gg, hh$ are generated
 similarly to the foregoing pixel-based value b, h by the use
 of the filter coefficients of Vector(1). Thereafter, using the
 filter coefficients of Vector(1), sum-of-products operation is
 10 made according to the following equation

$$j = c c - 5 d d + 20 h + 20 m - 5 e e + f f \quad \dots\dots (4)$$

or the following equation

$$j = a a - 5 b b + 20 b + 20 s - 5 g g + h h \quad \dots\dots (5)$$

Thereafter, a clip process is made according to the following
 15 equation, thus carrying out computation.

$$j = C l i p 1 \left((j + 512) \gg 10 \right) \quad \dots\dots (6)$$

Meanwhile, the pixel-based values $a, c, d, n, f, i, k,$
 q can be computed by a linear interpolation with a pixel-based
 values at integer-pixel accuracy (corresponding to the hatching
 20 in the figure) and a pixel-based value $G, H, UB1, h, j, m, s$
 at 1/2-pixel accuracy, as given in the following equation.

$$a = (G + b) >> 1$$

$$c = (H + b) >> 1$$

$$d = (G + h) >> 1$$

$$n = (M + h) >> 1$$

$$f = (b + j) >> 1$$

$$i = (h + j) >> 1$$

$$k = (j + m) >> 1$$

$$q = (j + s) >> 1 \quad \dots\dots (7)$$

Meanwhile, the pixel-based values e, g, p, can be computed by a linear interpolation using only the pixel-based values b, h, m, s at 1/2-pixel accuracy, as given in the following equation.

$$e = (b + h) >> 1$$

$$g = (b + m) >> 1$$

$$p = (h + s) >> 1 \quad \dots\dots (8)$$

5

Furthermore, the pixel-based value r is a point called "interpolatively challenging position" under the JVT coding scheme, which can be computed as in the following equation.

$$r = (G + H + M + N + 2) >> 2 \quad \dots\dots (9)$$

10

Now, the motion compensating process at 1/8-pixel accuracy

(interpolation process) is explained with using Fig. 2.

Under the JVT coding scheme, the following equation is defined as a polyphenol filter coefficient to generate a pixel-based value (luminance signal) at 1/4-pixel accuracy.

$$\begin{aligned}
 &(-3, 12, -37, 229, 71, -21, 6, -1) \\
 &(-3, 12, -39, 158, 158, -39, 12, -3) \\
 &(1, -6, -21, 71, 229, -37, 12, -3) \\
 &\dots\dots (10)
 \end{aligned}$$

5

Concerning the pixel-based values b^h , b^v at 1/4-pixel accuracy, the filter coefficients in Equation (10) are used to make a sum-of-products operation with a pixel-based value A at integer-pixel accuracy as an input sample. Thereafter, a clip process is made according to the following equation, thereby carrying out computation.

10

$$b^{h,v} = \text{Clip1}(b + 128) \gg 8 \quad \dots\dots (11)$$

Meanwhile, concerning the pixel-based value c^m , c^q , a sum-of-products operation is first carried out on the pixel-based value b^h , b^v . Then, using a value not to make a clip process, a sum-of-products operation is further made with the pixel-based value b^h , b^v as an input. Thereafter, a clip process is made according to the following equation, thereby carrying out computation.

15

$$c^{m,q} = \text{Clip1}(c + 32768) \gg 16 \quad \dots\dots (12)$$

Meanwhile, the pixel-based value d at 1/8-pixel accuracy can be computed depending upon its position by a linear interpolation according to the following equation.

$$d = (A + b^h) \gg 1$$

$$d = (A + b^v) \gg 1$$

$$d = (b^h + c^q) \gg 1$$

$$d = (b^v + c^q) \gg 1$$

$$d = (c^q + c^q) \gg 1$$

$$d = (b^v + b^v) \gg 1$$

$$d = (c^q + c^m) \gg 1 \quad \dots\dots (13)$$

5

Meanwhile, the pixel-based value e can be computed by using the pixel-based value b^h , b^v positioned on a diagonal line, according to the following equation.

$$e = (b^h + b^v) \gg 1 \quad \dots\dots (14)$$

10

Meanwhile, the pixel-based value g can be computed according to the following equation.

$$g = (A + 3c^m + 2) \gg 2 \quad \dots\dots (15)$$

Furthermore, the pixel-based values f^h , f^v can be computed

by the following equations.

$$f^h = (3b^h + b^v + 2) \gg 2$$

$$f^v = (3b^v + b^h + 2) \gg 2 \quad \dots\dots (16)$$

Herein, Fig. 3 shows a concept figure on the technique to generate a pixel-based value at 1/8-pixel accuracy by linear interpolation using a pixel-based value existing on the diagonal line.

Now, explained is the motion compensating process (interpolation process) of a chrominance signal under the JVT coding scheme, by using Fig. 4. Under the JVT coding scheme, in both cases of compensation for a motion at 1/4-pixel accuracy, and motion at 1/8-pixel accuracy, the chrominance signal at decimal-pixel accuracy can be computed by linear interpolation according to the following equation.

$$v =$$

$$\frac{(s-d^x)(s-d^y)A + d^x(s-d^y)B + (s-d^x)d^yC + d^x d^y D + s^2/2}{s^2}$$

$$\dots\dots (17)$$

Meanwhile, there are a number of motion compensating processes defined under the JVT scheme, as described above. It is important, in generating image compression information at

high compression ratio, to select an optimal one of the relevant modes for the macro-block, i.e., to search for a motion vector. For this reason, a motion vector is searched for by RD optimization.

5 Now, such RD optimization is explained in detail.

In RD optimization, provided that the search result is $J(m, \lambda_{\text{MOTION}})$, the motion vector is $m = (m_x, m_y)^T$, the predictive motion vector is $SA(T)D(s, c(m))$, $p = (p_x, p_y)^T$, the Lagrange multiplier for the motion vector is λ_{MOTION} , and the generated
10 information amount of a motion vector difference determined by a generation code amount table is $R(m - p)$, then the motion search at every pixel accuracy in motion compensation can be determined according to the following equation.

$$\begin{aligned} J(m, \lambda_{\text{MOTION}}) \\ = SA(T)D(s, c(m)) + \lambda_{\text{MOTION}} \cdot R(m - p) \end{aligned} \quad \dots\dots (18)$$

15 Herein, under the JVT coding scheme, there are defined two kinds of entropy coding, i.e., the method based on UVLC (Universal Variable Length Coding) and the method based on CABAC (Context-based Adaptive Binary Arithmetic Coding). For the generated information amount, ULVC is applied even where CABAC
20 is being applied. The residual energy (distortion) part of motion vector (corresponding to $SA(T)D(s, c(m))$ in Equation 18) can be determined by the following equation, provided that the

current-frame image signal is s and the reference-frame image signal is c .

$$SAD_c(s, c(m)) =$$

$$\sum_{x=1, y=1}^{B, B} |s(x, y) - c(x - m_x, y - m_y)| \quad \dots (19)$$

Note that, when correcting a motion vector at half-pixel accuracy or lower, SATD is used because determination is possible by the use of a Hadamard transform instead of a discrete cosine transform.

The Lagrange multiplier (λ_{MOTION}), provided that the quantization parameter is QP , for the I and P frames is given as in the following equation.

$$\lambda_{\text{MODE, P}} = \sqrt{0.85 * 2^{QP/3}} \quad \dots (20)$$

Meanwhile, for the B frame is given as in the following equation.

$$\lambda_{\text{MODE, B}} = \sqrt{4 * 0.85 * 2^{QP/3}} \quad \dots (21)$$

15

Now, explained is the case of selecting a reference frame image to be actually referred from a group SF of a plurality of reference frame images (Fig. 17) in RD optimization. In RD optimization, provided that, concerning selecting a reference frame image, the result of that selection is $J(\text{REF} | \lambda_{\text{MOTION}})$ and the value the amount of generated information on the reference

frame is determined by UVLC is $R(REF)$, selected is a reference frame image having a value at which the relevant selection result $(J(REF | \lambda_{MOTION}))$ assumes a minimum according to the following equation.

$$\begin{aligned}
 & J(REF / \lambda_{MOTION}) \\
 &= SATD(s, c(REF, m(REF)) \\
 &+ \lambda_{MOTION} \cdot (R(m(REF) - p(REF)) + R(REF)) \\
 & \dots\dots (22)
 \end{aligned}$$

5

Meanwhile, concerning to select a predictive direction of $N \times M$ blocks in the B frame, provided that the result of that selection is $J(PDIR | \lambda_{MOTION})$, selected is a predictive direction of $N \times M$ blocks having a value at which the relevant selection result $(J(PDIR | \lambda_{MOTION}))$ assumes a minimum according to the following equation.

10

$$\begin{aligned}
 & J(PDIR / \lambda_{MOTION}) \\
 &= SATD(s, c(PDIR, m(PDIR)) \\
 &+ \lambda_{MOTION} (R(m(PDIR) - p(PDIR)) \\
 &+ R(REF(PDIR))) \\
 & \dots\dots (23)
 \end{aligned}$$

Now, explained is macro-block mode selection mentioned above in Fig. 16, in RD optimization. In RD optimization,

provided that, concerning selecting a macro-block mode, the result of that selection is $J(s, c, \text{MODE} | \text{OP}, \lambda_{\text{MOTION}})$, selected is a macro-block mode having a value at which the relevant selection result ($J(s, c, \text{MODE} | \text{OP}, \lambda_{\text{MOTION}})$) assumes a minimum according to the following equation.

$$\begin{aligned} & J(s, c, \text{MODE} | \text{QP}, \lambda_{\text{MOTION}}) \\ & = \text{SSD}(s, c, \text{MODE} | \text{QP}) \\ & + \lambda_{\text{MOTION}} \cdot R(s, c, \text{MODE} | \text{QP}) \end{aligned} \quad \text{.....(24)}$$

The MODE, to assume a selection candidate for the macro-block mode, is shown based on each frame type in Table 1.

$$\begin{aligned} \text{I f r a m e : M O D E} & \in \left\{ \text{INTRA4x4, INTRA16x16,} \right\} \\ \text{P f r a m e : M O D E} & \in \left\{ \begin{array}{l} \text{INTRA4x4, INTRA16x16, SKIP,} \\ \text{16x16, 16x8, 8x16, 8x8} \end{array} \right\} \\ \text{B f r a m e : M O D E} & \in \left\{ \begin{array}{l} \text{INTRA4x4, INTRA16x16, DIRECT,} \\ \text{16x16, 16x8, 8x16, 8x8} \end{array} \right\} \end{aligned}$$

Table 1

In Table 1, "SKIP" has a size of vertical/horizontal 16x16 pixels and represents vector residues and those having coefficient residues not to be sent toward the decode side. "SSD" is a value of error sum of squares represented by the following equation provided that the macro-block generated information amount when MODE and QP are selected is $R(s, c, \text{MODE} | \text{QP})$, the luminance components of the re-configured and original images are $c_y[x, y, \text{MODE} | \text{QP}]$ and $s_y[x, y]$, and the chrominance components are c_u, c_v and s_u, s_v .

$$\text{SSD}(s, c, \text{MODE} | \text{QP}) =$$

$$\sum_{x=1}^{16} \sum_{y=1}^{16} (s_y[x, y] - c_y[x, y, \text{MODE} | \text{QP}])^2 +$$

$$\sum_{x=1}^8 \sum_{y=1}^8 (s_u[x, y] - c_u[x, y, \text{MODE} | \text{QP}])^2 +$$

$$\sum_{x=1}^8 \sum_{y=1}^8 (s_v[x, y] - c_v[x, y, \text{MODE} | \text{QP}])^2$$

..... (25)

10

Note that the amount of generated information includes that corresponding to all the pieces of information, such as headers, motion vectors and orthogonal transform coefficients. Meanwhile, the Lagrange multiplier (λ_{MOTION}) is provided as in the Equations (20) and (21).

15

The sub-macro block mode mentioned in Fig. 16 is

selection-processed similarly to the case on the macro-block mode. The MODE, to assume a selection candidate for the sub-macro block mode, is shown based on each frame type in Table 2.

$$P f r a m e : M O D E \in \left\{ \begin{array}{l} I N T R A 4 A \times 4, \\ 8 \times 8, 8 \times 4, 4 \times 8, 4 \times 4 \end{array} \right\}$$

$$B f r a m e : M O D E \in \left\{ \begin{array}{l} I N T R A 4 \times 4, D I R E C T, \\ 8 \times 8, 8 \times 4, 4 \times 8, 4 \times 4 \end{array} \right\}$$

5

Table 2

In case the foregoing processes as defined under the JVT coding scheme be applied to the image coding apparatus, a huge amount of operation is needed upon searching for motion vectors.

10 Accordingly, the present invention reduces the amount of operation possibly without losing the compression efficiency.

Now, one embodiment of the present invention is described in detail with reference to the drawings.

15 Image Coding apparatus Configuration

Fig. 5 is a block diagram of the overall configuration of an image coding apparatus 1 to which the JVT coding scheme is applied.

In the image coding apparatus 1, the motion image signal

S1 supplied from an external source is generated into motion image data D1 through an A/D converting section 2, followed by being temporarily stored in an image rearranging buffer 3. In the image rearranging buffer 3, the motion image data D1 are
5 rearranged in a compression-coding order on a frame-image unit basis. The rearranged frame images are sequentially forwarded as frame data D2 to an adder 4 and motion-prediction compensating section 5.

The motion-prediction compensating section 5, although
10 detailed later, divides the frame image of the frame data D2 provided from the image rearranging buffer 3 into motion compensating blocks having a size of vertical/horizontal 16×16 pixels, the greatest pixel-based size of the motion-compensating blocks mentioned in Fig. 16 (hereinafter, this is referred to
15 as the maximum motion-compensating block). Based on each maximum motion-compensating block, optimal motion vectors D3 are respectively detected by using a plurality of reference frame images while sequentially changing the pixel-based size of the motion-compensating block. These are sequentially forwarded
20 to a reversible coding apparatus 18.

Meanwhile, the motion-prediction compensating section 5 sequentially forwards to the adders 4 and 11 a plurality of predictive data D4 obtained by compensating for motion according to a plurality of motion vectors D3.

25 In the case that predictive data D4 is provided from the

motion-prediction compensating section 5, the adder 4 subtracts a plurality of predictive data D4 corresponding to the relevant frame data D2 from the respective frame data D2 (image type other than I picture) supplied from the image rearranging buffer 6, thereby generating difference data D5 and forwarding them to an orthogonal transform section 6.

On the contrary, when predictive data D4 is not provided, the adder 4 forwards the frame data D2 (I picture) provided from the image rearranging buffer 6, as difference data as they are, to the orthogonal transform section 6.

The orthogonal transform section 6 carries out an orthogonal transform process, such as discrete cosine transform, on the difference data D5 to thereby generate orthogonal transform coefficient data D6, and forwards them to a quantizing section 7.

The quantizing section 7 carries out a quantizing process on the orthogonal transform coefficient data D6, to thereby generate quantized data D7 and forward them to the reverse quantizing section 9 and reversible coding section 18.

In this case, in the quantizing section 7, the quantization control data D8 for determining a quantization parameter value are provided under feedback control by a rate control section 8, thereby generating quantized data D7 in accordance with a quantization parameter value of the quantization control data D8.

The reversible coding section 18 carries out a reversible coding process conforming, for example, to UVCL on the quantized data D7 provided from the quantizing section 7 and the motion vector D3 provided from the motion-prediction compensating section 5, thereby generating coded data D15 and outputting them through a storage buffer 19.

On the other hand, the reverse quantizing section 9 carries out a reverse quantizing process on the quantized data D7 provided from the quantizing section 7, to thereby restore orthogonal transform coefficient data D9 corresponding to the orthogonal transform coefficient data D6 and forward them to the reverse orthogonal transform section 10.

The reverse orthogonal transform section 10 carries out a reverse orthogonal transform process on the orthogonal transform coefficient data D9 provided from the reverse quantizing section 9, to thereby restore difference data D10 corresponding to the difference data D5 and forward them to an adder 11.

The adder 11, when provided predictive data D4 from the motion-prediction compensating section 5, adds a plurality of predictive data D4 corresponding to the relevant difference data D10 to the difference data D10 provided by the reverse orthogonal transform section 11, thereby restoring frame data D11 corresponding to the frame data D2 and forwarding them to a deblock filter 12.

On the contrary, the adder 11, when not provided motion compensating data D4 from the motion-prediction compensating section 5, forwards the difference data D10 provided from the reverse orthogonal transform section 11, as frame data D11 as they are, to the deblock filter 12.

The deblock filter 12, when a distortion is caused between adjacent macro-blocks of among the frame data D11 provided from the adder 11, smoothens the relevant distorted part by filtering and stores, as required, the smoothed frame data D11, as reference frame data D12 corresponding to one reference frame image, to a frame memory 13.

On the contrary, the deblock filter 12, when no distortion is caused between adjacent macro-blocks of among the frame data D11, stores as required the frame data D11, as reference frame data D12, to a frame memory 13.

In this manner, the image coding apparatus 1 is configured to store the reference frame data D12, in an amount of data corresponding to a plurality of reference frame images to be taken as reference candidates, to the frame memory 13.

In addition to the configuration, a first thinning-out section 14 thins out by a half horizontally vertically the pixels of a reference frame image of the reference frame data D12 read from the frame memory 13, thereby generating a size-reduced image having a 1/4 reduction ratio with respect to the reference frame image as an original image (hereinafter, this is referred to

as a 1/4 size-reduced reference image) and storing it as first size-reduced data D13 to a 1/4 frame memory 15.

In this case, the first thinning-out section 14 previously stores a FIR filter coefficient value defined in the foregoing Equation (1) in its internal memory (not shown). Using the FIR filter coefficient value, the pixels are thinned out according to a process similar to the motion-compensating process mentioned in Fig. 1. This can generate a 1/4 size-reduced reference image.

Meanwhile, the first thinning-out section 14 is configured, when the reference frame data D12 is an interlaced operated image, to make a thinning-out every field with respect to a vertical direction as shown in Fig. 6. For the first field, filtering is made by the use of a filter coefficient value given by the following equation and previously stored in the internal memory.

$$\left\{ -29, 0, 88, 138, 88, 0, -29 \right\} // 256 \quad \dots\dots (26)$$

For the second field, filtering is made by the use of a filter coefficient value given by the following equation previously stored in the internal memory.

$$\left\{ 1, 7, 7, 1 \right\} // 16 \quad \dots\dots (27)$$

Due to this, pixel thinning-out can be done every other of pixel-based values inputted.

A second thinning-out section 16 previously stores in its internal memory a FIR filter coefficient value defined in the foregoing Equations (1), (26) and (27). Similarly to the first thinning-out section 14, thinning out is made a half horizontally and vertically on the pixels of a 1/4 size-reduced reference image of first size-reduced data D13 read out of the 1/4 frame memory 15. Due to this, generated is a size-reduced image having a 1/16 reduction ratio (hereinafter, this is referred to as a 1/16 size-reduced reference image) with respect to the reference frame image as an original image. This is stored as second size-reduced reduced data D14 to the 1/16 frame memory 17.

In this manner, the image coding apparatus 1 generates 1/4 size-reduced and 1/16 size-reduced images respectively having 1/4 and 1/16 reduction ratios on a plurality of reference frame images corresponding to a plurality of ones of frame data D12 stored in the frame memory 13, in the number corresponding respectively to the plurality of reference frame images. These are stored to the corresponding 1/4 frame memory 15 or 1/16 frame memory 17.

Motion-Prediction Compensating Section Configuration

Now, explanation is made on the configuration of the motion-prediction compensating section to which the invention

is applied.

As shown in Fig. 7, the motion-prediction compensating section 5 as a motion-prediction compensating apparatus is configured by a pixel-block dividing section 20, a size-reduced block generating section 21 as hierarchizing means, and a motion-prediction compensating section 22 as search-range determining means and detecting means, to input the frame data D2 provided from the image rearranging buffer 3 to the pixel-block dividing section 20.

The pixel-block dividing section 20 divides the frame image of the frame data D2, based on the block unit having a vertical horizontal 16×16 pixel-based size to thereby generate a plurality of maximum motion compensating blocks, and sequentially forwards the maximum motion compensating blocks as maximum motion compensating block data D20 to the size-reduced block generating section 21 and motion-prediction compensating section 22.

The first size-reduced block generating section 21a of the size-reduced block generating section 21 previously stores a FIR filter coefficient value as defined in the foregoing Equations (1), (26) and (27) in its internal memory (not shown), and carries out a similar process to the first thinning-out section (Fig. 5) thereby thinning out the pixels.

In this case, the first size-reduced block generating section 21a horizontally vertically thins out a half the pixels of the maximum motion compensating block of the maximum motion

compensating block data D20 to thereby generate a block having a size of vertical/horizontal 8×8 pixels with a $1/4$ size-reduction ratio (hereinafter, this is referred to as a $1/4$ size-reduced block) on the maximum motion compensating block
5 having a size of vertical/horizontal 16×16 pixels, and forwards it as $1/4$ size-reduced block data D21 to a second size-reduced block generating section 21b and motion-prediction compensating section 22.

The second size-reduced block generating section 21b
10 previously stores a FIR filter coefficient value as defined in the foregoing Equations (1), (26) and (27) in its internal memory (not shown), and carries out a similar process to the second thinning-out section (Fig. 5) thereby thinning out the pixels.

In this case, the second size-reduced block generating
15 section 21b horizontally vertically thins out a half the pixels of the $1/4$ size-reduced block of the $1/4$ size-reduced block data D21 to thereby generate a block having a size of vertical/horizontal 4×4 pixels with a $1/16$ size-reduced ratio (hereinafter, this is referred to as a $1/16$ size-reduced block)
20 on the maximum motion compensating block having a size of vertically horizontally 16×16 pixels, and forwards it as $1/16$ size-reduced block data D22 to the motion-prediction compensating section 22.

The motion-prediction compensating section 22
25 recognizes, as an original image group on the uppermost layer

TS, the maximum motion compensating block UB1 corresponding to the maximum motion compensation block data D20 provided from the pixel-block dividing section 20 and the reference frame images SF1 - SF4 corresponding to, for example, four kinds of reference frame data D12 read out of the frame memory 13, as shown in Fig. 8.

Meanwhile, the motion-prediction compensating section 22 recognizes, as a size-reduced image group on the first lower layer BSf, the 1/4 size-reduced block CBf corresponding to the 1/4 size-reduced block data D21 provided from the first size-reduced block generating section 21a and the 1/4 size-reduced reference frame images CB1 - CB4 corresponding to, for example, four kinds of first size-reduced image data D13 read out of the 1/4 frame memory 15.

Furthermore, the motion-prediction compensating section 22 recognizes, as a size-reduced image group on the lowermost second layer BSS, the 1/16 size-reduced block CBs corresponding to the 1/16 size-reduced block data D22 provided from the second size-reduced block generating section 21b and the 1/16 size-reduced reference frame images CB11 - CB14 corresponding to the second size-reduced image data D14 read out of the 1/16 frame memory 17.

Thus, the motion-prediction compensating section 22 carries out a motion-prediction compensating process by the use of the recognized image groups of the uppermost layer TS, first

lower layer BSf and second lower layer BSs to thereby generate a motion vector D3 and predictive data D4, and forwards the motion vector D3 to the reversible coding section 18 and the predictive data D4 to the adders 4 and 11.

5 In this manner, the motion-prediction compensating section 5 carries out a motion-prediction compensating process with separation to the hierarchical layers TS, BSf and BSs, by taking a set of the maximum motion compensating block UB1, 1/4 size-reduced block CBf and 1/16 size-reduced block CBs and the
10 reference frame image SF1 - SF4, 1/4 size-reduced reference image CB1 - CB4 and 1/16 size-reduced reference image CB11 - CB14 respectively corresponding to the reduction ratios of the maximum motion compensating block UB1, 1/4 size-reduced block CBf and 1/16 size-reduced block CBs.

15 Meanwhile, the motion-prediction compensating section 22 is configured to carry out a motion-prediction compensating process by a control section configured by a not-shown CPU, ROM, RAM and the like, according to a predetermined motion-prediction compensating program previously stored in the ROM. The
20 motion-prediction compensating process procedure is explained below.

Motion-prediction Compensating Process Procedure

As shown in Fig. 9, the motion-prediction compensating
25 section 22 moves from the start step of a routine RT1 to the

following step SP1, to sequentially make a block matching on the respective ones of 1/16 size-reduced reference images CB11 - CB14 on the lowermost, second lower layer BSs (Fig. 8) with a 1/16 size-reduced block CBs having a size of vertical/horizontal 4x4 pixels, thereby searching for a motion vector on the second lower layer BSs (hereinafter, this is referred to as a second lower layer motion vector).

Specifically, the motion-prediction compensating section 22 determines only the absolute-value total sums of differences between every pixel-based value within the 1/16 size-reduced block CBs and the pixel-based value of a block within the 1/16 size-reduced reference image CB11 (CB12 - CB14) corresponding to that block CBs (hereinafter, this is referred to as a corresponding block) without applying RD optimization, for example, as shown in Fig. 10. Based on a motion amount between the corresponding block CTB of among those the absolute-value total sum assumes the minimum and the 1/16 size-reduced block CBs, detected is a second lower layer motion vector 1/16MV1 (1/16MV2 - 1/16MV4) on the second lower layer BSs (Fig. 8). Then, the process moves to the next step SP2.

In this case, the motion-prediction compensating section 22 makes a block matching on the 1/16 size-reduced reference image CB11 - CB14 smaller in search range as compared to the reference frame image SF1 - SF4 as an original image without making reference to a generation code amount table representative

of a generation code amount of a motion vector difference and without carrying out an operation of quantizing function (Lagrange multiplier on the motion vector). This can reduce the process burden upon determining the second lower layer motion
5 vectors $1/16MV1 - 1/16MV4$.

In the step SP2, the motion-prediction compensating section 22 makes a motion compensation process (interpolation process) on the horizontal and vertical components of the second lower layer motion vector $1/16MV1 - 1/16MV4$ detected in the step
10 SP1 as shown for example in Fig. 11, thereby enlarging the horizontal component twice and the vertical component twice, i.e. four times in total.

In this case, the motion-prediction compensating section 22, even when the $1/4$ -pixel accuracy (Fig. 1) or $1/8$ -pixel accuracy (Fig. 2) under the JVT coding scheme is selected for
15 the maximum motion compensating block UB1 on an original image corresponding to the $1/16$ size-reduced block CBs, makes a motion compensation according to an integer-pixel accuracy or $1/2$ -pixel accuracy based on linear interpolation. This can reduce the
20 process burden during the motion compensation.

The motion-prediction compensating section 22 determines the range UAR1 (UAR2 - UAR4) including the peripheral several pixels in a rectangular range AR having apexes of start and end points of the second lower layer motion vector enlarged by the
25 motion compensation, as a search range within a $1/4$ size-reduced

reference image CB1 (CB2 - CB4) on the first lower layer BSf (hereinafter, this is referred to as a first lower layer search range). The process moves to the next step SP3.

In the step SP3, the motion-prediction compensating section 22 sequentially makes a block matching on the first lower layer search ranges UAR1 - UAR4 within the 1/4 size-reduced reference images CB1 - CB4 determined in the step SP2 with a 1/4 size-reduced block CBf having a size of vertical/horizontal 8x8 pixels instead of applying RD optimization similarly to the process in the step SP1, thereby detecting motion vectors 1/4MV1 - 1/4MV4 (Fig. 8) (hereinafter, this is referred to as a first lower layer motion vector) in the first lower layer. The process moves to the next step SP4.

In the step SP4, the motion-prediction compensating section 22 carries out a similar process to the process in the step SP2 on the first lower layer motion vectors 1/4MV1 - 1/4MV4 detected in the step SP3, thereby determining a search range (hereinafter, this is referred to as an uppermost layer search range) TAR1 - TAR4 within the reference frame image SF1 - SF4 (Fig. 8) on the uppermost layer TS. The process moves to the next step SP5.

In the step SP5, the motion-prediction compensating section 22 sequentially makes a block matching by RD optimization on the first uppermost layer search ranges TAR1 - TAR4 (Fig. 8) determined in step SP4, with a motion compensating block UB1

having a size of vertical/horizontal 16×16 pixels as defined under the JVT coding scheme, thereby detecting a motion vector related to the motion compensating block UB1. The process moves to the next step SP6.

5 In this case, the motion-prediction compensating section 22 searches only within the first uppermost layer search range TAR1 - TAR4 and detects a motion vector related to the motion compensating block UB1 without searching for a motion vector throughout the entire maximum motion compensating block as
10 defined under the JVT coding scheme, thereby making it possible to greatly reduce the process burden upon RD optimization requiring a huge operation amount as noted before.

 In step SP6, the motion-prediction compensating section 22 uses the first uppermost layer search range TAR1 - TAR4 on
15 the uppermost layer TS (Fig. 8), to carry out a block matching by RD optimization on each motion compensating block UB2 - UB34 while sequentially changing from the greater pixel-based size to the smaller pixel-based size based on each of the first to third series SS1 - SS3 whose motion compensating blocks are the
20 same in block form as shown in Fig. 12.

 In actuality, the motion-prediction compensating section 22, in the first series SS1, carries out a similar process to the search range determination noted in Fig. 11 (hereinafter, this is referred to as a search range determining process)
25 depending upon a motion vector of motion compensating block UB1

detected in the step SP5, to determine a second uppermost layer search range TAR5 - TAR8 smaller than the first uppermost layer search range TAR1 - TAR4 (Fig. 8) as shown, for example, in Fig. 13. Within only the relevant second uppermost layer search range TAR5 - TAR8, detected are two kinds of motion vectors related to the motion compensating blocks UB2 and UB3 each having a size of vertical/horizontal 8×16 pixels.

Then, the motion-prediction compensating section 22 carries out a search-range determining process depending upon a motion vector related to the motion compensating block UB2 and UB3, to determine a third uppermost layer search range TAR9 - TAR12 smaller than the second uppermost layer search range TAR5 - TAR8. Within only the relevant third uppermost layer search range TAR9 - TAR12, detected are four kinds of motion vectors related to the motion compensating blocks UB11 - UB14 each having a size of vertical/horizontal 4×8 pixels.

Meanwhile, the motion-prediction compensating section 22, in the second series SS2, detects two kinds of motion vectors related to the motion compensating blocks UB4 and UB5 each having a size of vertical/horizontal 16×8 pixels only within the second uppermost layer search range TAR 5 - TAR8 similarly to the process on the first series SS1, nearly simultaneously to the process on the first series SS1. Thereafter, within only the relevant third uppermost layer search range TAR9 - TAR12, detected are four kinds of motion vectors related to the motion compensating

blocks UB15 - UB18 each having a size of vertical/horizontal 8x4 pixels.

Furthermore, the motion-prediction compensating section 22, in the third series SS3, detects four kinds of motion vectors related to the motion compensating blocks UB6 - UB9 each having a size of vertical/horizontal 8x8 pixels only within the second uppermost layer search range TAR5 - TAR8 similarly to the process on the first series SS1, nearly simultaneously to the process on the first series SS1. Thereafter, within only the relevant third uppermost layer search range TAR9 - TAR12, detected are motion vectors related to the motion compensating blocks UB19 - UB34 each having a size of vertical/horizontal 4x4 pixels.

In this manner, the motion-prediction compensating section 22 reduces its search range each time sequentially changing to a motion compensating block of a smaller pixel-based size, thereby making possible to reduce the process burden upon detecting a motion vector in a smaller pixel-based size of motion compensating block requiring a greater operation amount by an increase of the number of blocks as compared to a motion compensating block greater in pixel-based size. Furthermore, process burden can be greatly reduced as compared to the case of determining a motion vector in each pixel-based size of motion compensating block (Fig. 16) throughout the entire macro-block as defined under the JVT coding scheme.

In this manner, the motion-prediction compensating

section 22 detects motion vectors respectively related to the remaining motion compensating blocks UB2 - UB34 other than the motion compensating block UB1. Thereafter, the process moves to the next step SP7.

5 In this connection, the motion-prediction compensating section 22, in the first to third series SS1 - SS3, used the same one of the second uppermost layer search range TAR5 - TAR8 and third uppermost layer search range TAR9 - TAR12. However, this is not limiting, i.e., in case the search range is reduced
10 each time sequentially changing to the smaller pixel-based size of motion compensating block, the form or the like may be different.

 In the step SP7, the motion-prediction compensating section 22 detects as the final motion vector D3 (Fig. 7) an
15 optimal motion vector capable of generating the smallest code amount among the motion vectors of the motion compensating blocks UB1 - UB34 respectively detected by RD optimization in the steps SP5 and SP6. The process moves to the next step SP8.

 In the step SP8, the motion-prediction compensating
20 section 22 carries out a 1/4 motion compensating process noted before for example in Fig. 1 according to the motion vector D3 detected in the step SP7, thereby generating predictive data D4 (Fig. 7). The process moves to the next step SP9.

 In the step SP9, the motion-prediction compensating
25 section 22 forwards the motion vector D3 detected in the step

SP7 to the reversible coding section 18 and the predictive data D4 generated in the step SP8 to the adders 4 and 11. The process moves to the next step SP10, to end the motion-prediction compensating process.

5

Operation and Effect

In the above configuration, the motion-prediction compensating section 5 takes as the uppermost layer TS a motion compensating block UB1 (Fig. 8) having the greatest pixel-based size among the motion compensating blocks defined under the JVT coding scheme. By thinning out over the maximum motion compensating block UB1, generated and hierarchized are a size-reduced block CBs on first lower layer having a 1/4 size-reduction ratio and a 1/16 size-reduced block CBf on second lower layer having a 1/16 size-reduction ratio.

Then, the motion-prediction compensating section 5 uses a plurality of reduced reference images CF11 - CF14 and CF1 - CF4 of the lower layers (first lower layer BSf and second lower layer BSs) excepting the uppermost layer TS and the reduced blocks CBs and CBf, to hierarchically search for motion vectors and determine a first uppermost layer search range TAR1 - TAR4 for motion vector search by RD-optimization on the uppermost layer TS (steps SP1 - SP4).

In this case, the motion-prediction compensating section 5 detects a motion vector by carrying out a block matching based

only on the absolute-value total sum of the difference from the size-reduced block without applying RD optimization which takes into account a generated code amount representative of a generated code amount of motion vector difference and a
5 quantizing function, and makes a motion-compensation on the detected motion vector by linear interpolation at integer-pixel accuracy or 1/2-pixel accuracy without making a motion-compensation at 1/4-pixel accuracy (Fig. 1) or 1/8-pixel accuracy (Fig. 2). Thus, a first uppermost layer search range
10 TAR1 - TAR4 can be determined by a reduced amount of operation.

In this state, the motion-prediction compensating section
5 detects motion vectors related respectively to the motion compensating blocks UB1 - UB34 (Fig. 12) having respective pixel-based sizes by the use of the first uppermost layer search
15 range TAR1 - TAR4. Among the detected motion vectors, detected is the optimal and final motion vector D3 capable of generating the smallest amount of code.

Accordingly, despite the motion-prediction compensating section 5 is added by the new operation in an amount required
20 in the process of up to determining the first uppermost layer search range TAR1 - TAR4, restricted is the search range for RD optimization requiring by far greater operation amount than that operation amount. Correspondingly, process burden can be greatly reduced as compared to the case to detect a motion vector
25 by searching for a motion vector throughout the entire

macro-block on each motion compensating block in each pixel-based size as defined under the JVT coding scheme.

In addition, the motion-prediction compensating section 5, as noted above in Fig. 12, when detecting motion vectors related to the motion compensating blocks UB1 - UB34 (Fig. 12) in the respective pixel-based sizes, reduces the search range each time of sequentially changing the motion compensating blocks from the greater pixel-based size to the smaller pixel-based size. This can reduce the process burden upon detecting a motion vector 10 on a motion compensating block in a smaller pixel-based size. As a result, the overall process burden can be reduced furthermore.

According to the above configuration, the motion compensating block UB1 in the greatest pixel-based size is 15 hierarchized. Using a plurality of reduced reference images CF11 - CF14 and CF1 - CF4 in the hierarchized first lower layer BSf and second lower layer BSs as well as reduced blocks CBs and CBf, the optimal motion vector in the uppermost layer TS is narrowed down to roughly determine a first uppermost layer 20 search range TAR1 - TAR4 by a reduced amount of operation. Thereafter, using the relevant search range TAR1 - TAR4, a motion vector is detected by RD optimization. This can greatly reduce the process burden upon detecting the motion vector D3.

25 Other Embodiment

Incidentally, the above embodiment described the case for carrying out a motion-prediction compensation by sequentially changing to the respective motion compensating blocks having respective pixel-based sizes of vertical/horizontal 16×16 pixels, vertical/horizontal 8×16 pixels, vertical/horizontal 16×8 pixels, vertical/horizontal 8×8 pixels, vertical/horizontal 4×8 pixels, vertical/horizontal 8×4 pixels and vertical/horizontal 8×4 pixels. However, the invention is not limited to this but motion-prediction compensation may be made by sequentially changing to the motion compensating blocks in other various pixel-based sizes, e.g., vertical/horizontal 16×4 pixel-based size.

Furthermore, although the above embodiment described the case that the invention is applied to the JVT coding scheme, this is not limitative. In brief, the invention can be applied to the other various coding schemes that motion-prediction compensation is made by using a plurality of reference frame images while sequentially changing the pixel-based size of the motion compensating block on each motion compensating block that the objective frame image to be processed is divided of the successive frame images.

Furthermore, although the above embodiment described the case that thinning-out is made using Equation (1) taking as a FIR filter coefficient, the invention is not limited to this but other various FIR filter coefficient values may be used to

generate a $1/2$ phase.

Furthermore, although the above embodiment described the case that reduction is at the same reduction ratio in the horizontal and vertical directions, the invention is not limited to this but size-reduction may be at different size-reduction ratios between the horizontal direction and vertical direction.

Furthermore, although the above embodiment described the case that size-reduction is at $1/4$ and $1/16$ size-reduction ratios, the invention is not limited to this, i.e., size-reduction may be at the other reduction ratio, e.g., 1 over the power of 2 .

Furthermore, the above embodiment described the case to determine a motion vector depending upon an absolute-value total sum of differences between every pixel-based value within a $1/16$ size-reduced block CBs and a pixel-based value of the corresponding block within the $1/16$ size-reduced reference image CB11 - CB14 corresponding to the relevant block CBs, or between every pixel-based value within a $1/4$ size-reduced block CBf and a pixel-based value of the corresponding block within the $1/4$ size-reduced reference image CB1 - CB4 corresponding to the relevant block CBf. However, the invention is not limited to this but a square sum of the differences may be determined. In brief, motion vectors can be determined on the basis of the difference values between the size-reduced block pixel-based value and the pixel-based value of reduced reference image.

Furthermore, the above embodiment described the case, as noted above in Fig. 8, of using every reference size-reduced image CF11 - CF14 and CF1 - CF4 in the second lower layer BSs and first lower layer BSf as well as size-reduced blocks CBs and CBf, to determine the first uppermost layer search range TAR1 - TAR4 on every reference frame image SF1 - SF4 in the uppermost layer TS. However, the invention is not limited to this but may determine any of the uppermost layer search ranges TAR1 - TAR4.

In this case, following the process of the step SP1 in the motion-prediction compensating process procedure mentioned in Fig. 9, added is a step of selecting, among reference size-reduced images CF11 - CF14 (Fig. 8), a reference size-reduced image CF13, for example, that the difference of pixel-based value is minimum from the reduced block CBs by a block matching with a size-reduced block CBs on the second lower layer BSs. Using only the reference size-reduced image CF3 on first lower layer BSf corresponding to the reference size-reduced image CF13 selected in that step, the motion-prediction compensating section 22 carries out the processes of steps SP2 to SP4, to thereby determine only the first uppermost layer search range TAR3.

In this manner, the motion-prediction compensating section 22, in the second lower layer BSs, selects a reference frame image SF3 for RD-optimization motion-vector search on the

uppermost layer TS. This can reduce the respective amounts of operation corresponding to a plurality of reference size-reduced images CF11, CF13, CF14, CF1, CF3 and CF4 and a plurality of reference frame images SF1, SF3 and SF4. The overall process
5 burden can be greatly reduced of up to detecting the final optimal motion vector D3 (Fig. 7).

Furthermore, where the motion compensating block UB1 to be processed is of a picture, the motion-prediction compensating section 22 in the second lower layer BSs may take, by a block
10 matching with the reduced block CBs, a predictive-direction mode having a motion vector the pixel-based value difference is minimum from the size-reduced block CBs as a predictive-direction mode for the motion compensating block UB1 among the reference reduced images CF11 - CF14 (Fig. 8), similarly to the foregoing
15 selection of a reference frame image SF3. By doing so, the overall process burden can be further reduced of up to detecting the final optimal motion vector D3 (Fig. 7).

Furthermore, the above embodiment described the case that, as described in Fig. 11 as an example of determination of search
20 ranges for example, the motion vector detected on the second lower layer is enlarged to determine as a first lower layer search range a range UAR1 including the peripheral several pixels of a rectangular range AR having, as apexes, the start and endpoints of that enlarged motion vector. However, the invention is not
25 limited to this. In brief, the peripheral pixel-based range

including an enlarged motion vector can determine a search range, e.g. trapezoidal range, rectangular-parallelepiped range, circular range including peripheral pixels having as apexes start and end points of that enlarged motion vector.

5 Furthermore, the above embodiment described the case to detect the optimal motion vectors for the motion compensating blocks UB2 - UB34 in the order mentioned in Fig. 12. However, the invention is not limited to this but may detect the optimal motion vectors for the motion compensating blocks UB2 - UB34
10 in the order as in Fig. 14 shown attached with the same references to the corresponding parts to Fig. 12.

Furthermore, the above embodiment described the case to make a block matching by the use of all the pixel-based values within the 1/16 size-reduced block CBs, 1/4 size-reduced block
15 CBf or maximum motion compensating block UB1. However, the invention is not limited to this but may make a block matching by the use of every other pixel-based value with respect to each of the horizontal and vertical directions within the 1/16 size-reduced block CBs, 1/4 size-reduced block CBf or maximum
20 motion compensating block UB1 as shown in Fig. 15. In this case, the process burden of upon carrying out a block matching can be reduced furthermore.

Furthermore, the above embodiment described the case to realize the processes in the motion-prediction compensating
25 section 22 by a motion compensating program. The invention is

not limited to this but the processes in part or the entire may be realized on the hardware means of exclusive integrated circuits or the like.

Furthermore, the above embodiment described the case to
5 realize a size-reducing process by the hardware-configured size-reduced block generating section 21 (Fig. 7). However, the invention is not limited to this but the size-reducing process in the size-reduced block generating section 21 may be realized by being built in a motion-prediction compensating program in
10 the motion-prediction compensating section 22.

Furthermore, the above embodiment described the case for the motion-prediction compensating section 5 as a motion-prediction compensator to implement a motion-prediction compensation by expanding in the internal RAM (not shown) a
15 motion-prediction compensating program stored in the internal ROM (not shown) according to the foregoing motion-prediction compensating process of Fig. 9. The invention is not limited to this but motion-prediction compensation may be implemented by installing a program medium storing the motion-prediction
20 compensating program.

The program medium may be realized by a semiconductor memory, a magnetic disk or the like, besides a packaged media such as a CD-ROM (Compact Disc-Read Only Memory) or DVD (Digital Versatile Disc). Meanwhile, the means for storing a program
25 to the program medium may utilize a wired or wireless radio

communicating medium, such as local area networks, the Internet or digital satellite broadcast. Storage may be through various interfaces, such as routers and MODEMs.

As described above, according to the present invention,
5 when compensating for motion prediction on each of motion compensating blocks formed by dividing an objective frame image of successive frame images by using a plurality of reference frame images while sequentially changing pixel-based sizes of the motion compensating blocks, pixels are thinned out of the
10 motion compensating block having a greatest one of the pixel-based size to be taken as an uppermost layer of among the pixel-based sizes, to generate and hierarchize a size-reduced block in a lower layer having a predetermined size-reduction ratio. Thereafter, motion vector search ranges are determined
15 respectively within the reference frame images, on the basis of the size-reduced blocks and a plurality of size-reduced reference images size-reduced corresponding respectively to the size-reduction ratios of the size-reduced blocks. An optimal one of the motion vectors is detected while sequentially changing
20 the pixel-based sizes of the motion compensating blocks by using the search ranges determined.

In this case, motion vectors can be detected only in a limited search range of the reference frame image without using the entire of the reference frame image. Thus, the process burden
25 upon code processing can be reduced while possibly avoiding the

lowering of image quality.

The present patent application contains subject matter related to that disclosed in Japanese Priority Document JP2002-350138, filed in the JPO on December 2, 2002, the entire
5 contents of which being incorporated herewith by reference.